Decentralised Authentication Protocol for Devices & Users to Access Private Network Services Using Blockchain

Praneeth Gosu, Rishik Tanguturu, Shashi Prakash Aenugutala, Tyson Baptist D Cunha, Kiran Manjappa
National Institute of Technology Karnataka
Surathkal, India
{praneethgosu322, rishik.tulaa, shashiprakash1729, dcunhatyson}@gmail.com, kiranmanjappa@nitk.edu.in

***Abstract*—With recent advancements in the Internet of things, challenges to secure devices and data related to devices have increased. Adversaries using different threats manage to clone/hack/tamper devices by hacking credentials stored in centralised databases. In this work, a decentralised approach using blockchain is proposed to check the authenticity of the device/user trying to access the services of the service provider network. The proposed method uses public and private blockchain networks and Physical Unclonable Function (PUF) to authenticate the device/user and to store their credentials. The decentralised application runs on Hyperledger Fabric, an open-source platform for building blockchain networks. The proposed protocol is tested and implemented in the physical testbed containing Raspberry Pi and Arduino Mega's.**

I. Introduction

A. Overview

Authentication enables service-providing networks to stay secure by permitting only authenticated users to access their network or protected resources. Any device or user can provide their unique credentials to get into the network, but the genuineness of the user can not be known unless there is a source that will verify the genuineness of the device or user. Getting proof of the existence of the device or user from an authorized point of contact can complete authentication. The traditional approach uses a centralised database [1] to store the device's credentials for authentication. The centralised database has a threat of a single point of failure (SPOF) [2], and adversaries can also hack or modify the credentials in the database. To overcome these threats, blockchain, a decentralised and immutable approach, comes to the rescue.

Blockchain is a decentralized digital ledger that records transactions securely and transparently. Blockchain is a chain of blocks where each block contains a cryptographic hash of the previous block, a timestamp, and transaction data of the current block. Once a block is added to the blockchain, the data inside the block cannot be modified, ensuring the integrity and immutability of the information stored on the network. This makes blockchain ideal for applications like cryptocurrency transactions, supply chain management, and voting systems. The decentralized nature of blockchain means that it operates on a peer-to-peer network, with no central authority controlling the network. This enhances security and helps prevent fraud, and avoids SPOF in the system. Overall, blockchain technology offers a secure and transparent way of recording and verifying transactions and is set to impact a range of industries in the future significantly.

Decentralised applications are more widely used due to their advantages over centralised applications. Blockchain technology provides security and immutability to the data and also builds trust between the parties involved in the network with the help of Peer to Peer system and consensus algorithm. There are mainly two types of blockchains, public and private. In public blockchain, the data is available to everyone, and anyone can participate in consensus by joining the network. The Private Blockchain [3] network is only restricted to permitted or allowed users who can view of modify data. The permission to allow a peer to participate in the network lies only for a certificate authority present in the network.

On the other hand, the service provider networks are the ones that provide services or resources to the users/devices requesting or in need. In order to avail services from the service provider network, a device/user should undergo authentication to verify the device/user's unique identity and genuineness. That's where Physically Unclonable Function (PUF) comes into the picture, which can generate a unique bit stream of information from the intrinsic properties of a hardware device. Using a PUF key and proposed protocol, we verify the device that's present in a hybrid network and then allow the device into the private network. Hybrid blockchain networks are those which possess properties of both private and public blockchain networks.

B. Physical Unclonable Function

A Physical Unclonable Functions (also called Physically Unclonable Function), PUF, is a function which generates a unique response for a given challenge (input) called as Challenge Response Pairs (CRPs). These CRPs act as a distinctive ID for semiconductor devices like microprocessors. PUFs are frequently based on distinctive physical variations that naturally occur during the fabrication of semiconductors [4]. Even the manufacturers of these devices cannot control these exact variations as it depends on a unique threshold voltage for each transistor on a chip [5]. Hence, the PUFs can be used to generate unique fingerprint for a device based on the device physical variants. PUFs are frequently employed in applications with high security needs, notably cryptography, and are commonly implemented in integrated circuits.

Generally, whenever a device tries to enter a network, they provide credentials and prove its authenticity. After authentication, if the device is proven authentic, certain authorizations and access for the services are given to the device/user. But, verifying the device/user every time with their proof of existence and originality could bring a high computation cost to the service-providing network. Hence, the system to verify device authenticity using hybrid and private blockchain is proposed where hybrid blockchain deals with storing the device credentials via manufacturer during registration, and private blockchain will store current authenticated device credentials and also previous attempts (if any) made by the current device to save the computation time during next authentication. PUF will be used as a unique fingerprint ID and a unique secret key (K) specific to device will also be used as device credentials.

II. Literature Survey

A. Background Work

The initial survey includes analysis of different blockchain platforms. Well-known Public Blockchain networks such as Ethereum [6] and Bitcoin [7] allow anyone to join the network as a peer, and anyone can register into the network provided the required details. Ethereum included smart contracts (a collection of self-verifying, self-executing, and tamper-resistant programs [8]) into the blockchain, with Ether as its cryptocurrency. This platform allows the developer to write smart contracts specific to the service by taking a fee for each transaction in terms of gas. As Ethereum allows to develop a public network, the data can be publicly available, which is a confidentiality issue.

Private blockchain networks do not allow unknown peers to enter the network, thus providing confidentiality to the data. Many platforms allow the development of a private blockchain network, such as Hyperledger Fabric (HLF) [9], funded by Linux Foundation, where one can design and deploy a network along with chain code (used as smart contracts in HLF). In order to adapt the system to specific use cases and trust models, it provides modular consensus mechanisms.

Similarly, IOTA [10] is an open-source cryptocurrency and distributed ledger created for the Internet of Things (IoT). Compared to distributed ledgers based on blockchain, it is more scalable because it uses a directed acyclic graph to store transactions. Transaction validation in IOTA does not involve miners. Instead, nodes that send out new network transactions must first authorise two older ones. The ability to issue transactions without charging a fee so makes micro-transactions possible. Using a coordinator node run by the IOTA Foundation, the network reaches consensus. Therefore, HLF & IOTA don't need a separate concept of miners to validate a transaction, but the peers in the network validate the transaction.

B. Related Works

This section discusses the literature review on different authentication protocols for IoT devices. In [1], authors have used low-cost hardware devices to generate secret IDs and authenticate them using the proposed communication protocol, which is proved to be rigid and robust against attacks. This paper shows how to recognise an IoT device using a PUF key.

In [11], authors have used Digital signatures, Hashing and SRAM PUF to generate a unique ID for a device and authenticate the device based on digital signature and update on the ledger. The protocol is scalable and robust but is computationally expensive.

In [12], authors have proposed a mutual authentication scheme for the Internet of Things (IoT) that uses Physically Unclonable Functions (PUF) and blockchain technology. The proposed scheme aims to enhance the security of IoT devices by utilizing PUF as a cryptographic key generator and blockchain for secure data storage and communication. The authors evaluate the proposed scheme and show improved security, privacy, and reliability compared to traditional authentication methods. This research contributes to developing secure authentication schemes for IoT systems and highlights the potential of combining PUF and blockchain technology for securing IoT devices.

In [13], the authors have proposed a authentication protocol PUFchain 2.0 which enhances the security and sustainability of the original PUFchain system by incorporating hardware-based PUFs for device authentication and secure data storage. The authors have shown that the use of PUFs provides robust protection against various attacks and enhances the system's privacy by preventing unauthorized access to patient data. Additionally, PUFchain 2.0 utilizes a consensus mechanism and cryptographic techniques to ensure secure and efficient data transfer between devices in the network.

In [14], authors have proposed a combination of symmetric cryptography and zero-knowledge proofs to securely authenticate devices and users in the network while preserving their privacy. The protocol also allows users to revoke access to their personal data anytime, providing an additional layer of control and security. The authors perform extensive simulations and evaluations of the proposed protocol, demonstrating its effectiveness in preserving privacy while providing secure authentication in IoT-AmI environments. This research highlights the importance of privacy-preserving authentication protocols in IoT-AmI systems and provides a promising solution for ensuring user privacy in these environments.

In [15], authors have proposed a decentralized framework for ensuring device authentication and data security in the Internet of Medical Things (IoMT). The authors claim that the existing authentication and security methods in the IoMT are insufficient, as they are centralized and susceptible to hacking, and therefore propose a decentralized framework. This framework uses a combination of blockchain and cryptography to ensure secure communication between devices and protect sensitive medical data. The authors described the design and implementation of the framework and evaluated its performance in terms of communication overhead, processing time, and memory usage. The results show that the decentralized framework provides a secure and efficient solution for device authentication and data security in the IoMT. The paper concludes that the proposed framework can be useful for ensuring the security and privacy of medical devices and data in the next generation of the IoMT.

In [16], authors have proposed a new authentication method for Internet of Things (IoT) devices using a combination of blockchain technology and Physical Unclonable Functions (PUF). The authors claim that existing authentication methods for IoT devices are vulnerable to attacks and lack the ability to provide a secure and efficient authentication process. To address this issue, they propose a blockchain-based authentication method that uses PUF to create a secure key for each device. The authors describe the design and implementation of the authentication method and evaluate its performance in terms of communication overhead, processing time, and security. The results show that the proposed authentication method provides a secure and efficient solution for IoT devices, improving existing methods. The paper concludes that blockchain-based authentication with PUF is a promising solution for the security and privacy of IoT devices in the Industry Internet of Things (IIoT).

In [17], the authors have shown an ownership transfer protocol which is based on public blockchain. Their work focuses on tracking and tracing of Integral Circuits (ICs) in Supply Chain Management. Each IC will get an unqiue ID generated using PUF which is stored in the public blockchain to verify IC's authenticity. The authors made deployment on Ethereum blockchain network. The work is mainly concentrated on ownership transfer, tracing and tracking but not on device/user authentication.

C. Outcome of Literature Review

By the protocols and methods discussed in the above survey, the observations are as follows:

- To the best of our knowledge, no work is done using private and hybrid blockchain networks to store and secure the data for device authentication.
- To the best of our knowledge, no work is done using private blockchain to store data after authentication.
- An article [1] has used a centralised server to store credentials that are vulnerable to attacks. If credentials are known to the adversary, then the entire authentication protocol fails.

The Literature review summary is shown in TABLE I

III. Proposed Methodology

The proposed work uses two blockchain networks to implement the decentralised authentication protocol. One is a hybrid blockchain network, and the other is a permissioned private blockchain network. Generally, the flow is like when a device/user comes to get into the private network for services, the client-application of the private blockchain network,i.e., fabric-client the API (Application Programming Interface) service, will check if the device/user is in the private blockchain network. Suppose the device/user has not yet entered into the private network; then the authentication process starts by fetching the data of the device/user from the hybrid blockchain network as well as from the device. The hybrid blockchain network is expected to have data of all the legal devices.

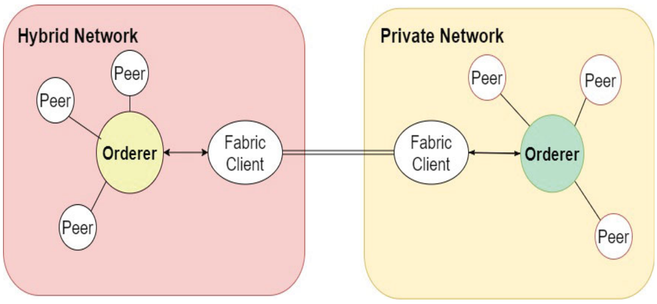


Fig. 1. Proposed System Architecture

A. System Architecture

The hybrid blockchain, is a blockchain network with properties of both public and private blockchain networks. It's a mix of both networks, hence called hybrid. This hybrid blockchain network contains properties of a private blockchain network; it only allows permissioned peers to perform consensus and gives editable data access to authorized users. This network resembles a public blockchain network by providing access to its data publicly. Both the blockchain networks will be developed using Hyperledger Fabric 2.4, which consists of fabric client. Fabric-client will be visible to the device/user so that the user-authentication process occurs here. The work also includes implementing the networks on a physical testbed setup consisting of Raspberry PI and Arduino devices. The role of Arduino devices comes in a while generating the PUF key (unique ID of a device generated by taking reading from SRAM) as mentioned in [1]. Fig.7 shows the sample testbed used in the paper, and it is planned to implement this paper and replicate the procedure to implement authentication protocol. The IoT devices will be registered in the public blockchain with the PUF key generated for the IoT device, which will be stored in the public blockchain. The hash of the transaction is stored in the IoT device so that it can be helpful when this device is trying to enter into a private blockchain network.

B. Device registration and authentication

Device registration phase is initiated when a device is manufactured by a manufacturer and the phase ends with the successful registration of the device in HYBNET.

1) Device Registration to HYBNET: PUF key-based key generation protocol [1] will generate a unique fingerprint ID for each device. The Manufacturer registers the device into the public blockchain network. The device registration phase is shown in Fig. 2. The manufacturer reads the device's PUF

TABLE I. LITERATURE REVIEW SUMMARY

Author	PKI	PUF	Centralised	Decentralised	Test-bed	Remarks
[1]	✗	✓	✓	✗	✓	Uses centralised secured database, vulnerable to attacks. They are not using error correction techniques for PUF key generation.
[11]	✓	✓	✗	✓	✓	Computationally expensive, scalable and robust.
[12]	✓	✓	✗	✓	✓	Effective in preventing unauthorized access by attackers
[13]	✓	✓	✗	✓	✗	Robust
[14]	✓	✓	✗	✓	✗	Provides a secure approach to authentication that prioritizes user privacy and security.
[15]	✓	✓	✗	✓	✗	Reliable and secure.
[16]	✓	✓	✗	✓	✗	Secure and efficient while maintaining a low computation and communication overhead.
[17]	✗	✓	✗	✓	✓	Public blockchain is used for deployment. Work is not concentrated on device/user authentication.

TABLE II. KEYWORDS DESCRIPTION

Keyword	Abbreviation
PUF	Physically Unclonable Function
SK	Secret Key
SPOF	Single Point of Failure
HLF	Hyperledger Fabric
IoT	Internet of Things
IC	Integrated Circuit
SRAM	Static random-access memory
MAC	Media Access Control
IoMT	Internet of Medical Things
IIoT	Industry Internet of Things
API	Application Programming Interface
HYBNET	Hybrid Blockchain Network
PVTNET	Private Blockchain Network
SDK	Software Development Kit
T_x	Transaction Hash
JWT	JSON Web Token
PKI	Public Key Infrastructure

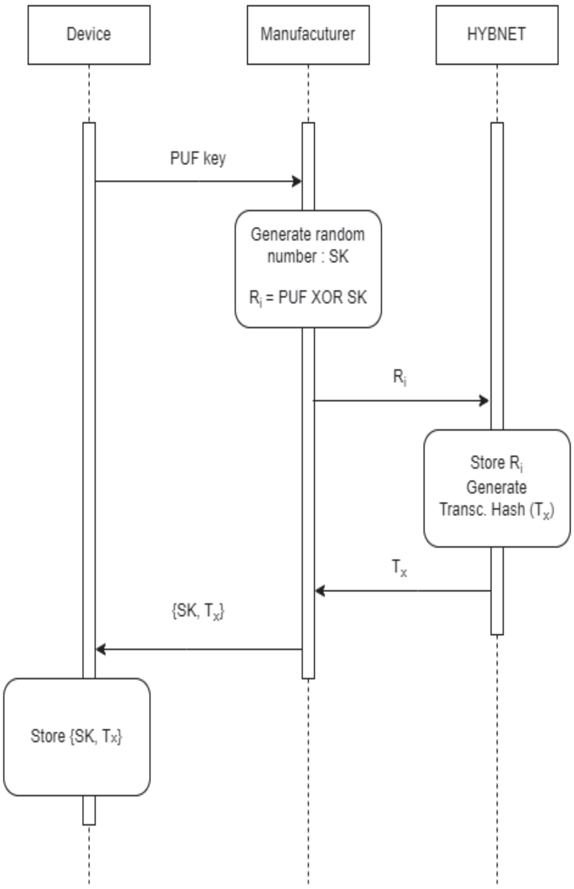


Fig. 2. Device Registration in HYBNET

key and generates a secret key (SK). Manufacturer computes R_i by using Eq. 1. The HYBNET will store R_i, the XOR of both the PUF key and secret key (SK), in its distributed ledger.

$$R_i = PUFKey \oplus SK \qquad (1)$$

Whenever the R_i is stored in the HYBNET, the transaction will produce a hash value called transaction hash (T_x). This T_x is stored in the device along with SK for further use.

2) Device Authentication: The authentication phase is initiated when a device requests or shows interest in joining the service provider network to avail services. The device authentication phase is shown in the Fig.3. Fabric-client-application is an API, a part of PVTNET based on Hyperledger Fabric. As shown in the Fig.3, When a device enters the private network, the device contacts the fabric client. The device provides its Secret Key (SK), Transaction Hash (T_x) and PUF key to the PVTNET Fabric client. When the device tries to get authenticated, there can be two cases. The first case is if the device is already registered in the PVTNET (or already authenticated previously), then the device will get authenticated directly by fetching data from PVTNET, and there will be no further communication with HYBNET. In the second case, if the device is not registered in the PVTNET (or first time authentication), the PVTNET's fabric client requests the HYBNET for R_i by sending T_x, which is provided by the device. HYBNET gives a response as $R^1{}_i$ to the fabric client if the device is registered with T_x in the HYBNET. If T_x is not present in HYBNET, then authentication fails since the device is considered as not registered by the manufacturer and considered as a fake or adversary. In the next step, the PVTNET's fabric client computes R_i using credentials (PUF Key and SK) submitted by the device during the first step. If R_i and $R^1{}_i$ match, authentication will be successful, and the device will get registered in the PVTNET, else if R_i and $R^1{}_i$ does not match, authentication is a failure.

IV. Experimental Setup

A. Building Blockchain Networks

The two blockchain networks, both hybrid and private blockchain networks, are designed with three organizations in each network. Each network consists of three organizations, three peers (one peer for each organization), one orderer, three CouchDB databases (one for each peer), and three certificate authorities (one for each organization). Organizations are groups that participate in the network for trust in data and services present in the blockchain.

There is one difference between HYBNET and PVTNET. The permissions of organizations present in HYBNET and PVTNET are different. The PVTNET is set such that the users who enter that network can change the data in it. But as for HYBNET, two sets of rules are included. One rule is view-only, where users only view the data, and the other rule is write the data so that manufacturers can add the device's R_i value. In HYBNET, an organization is solely dedicated to the users who opt for view-only and can register under this organization only.

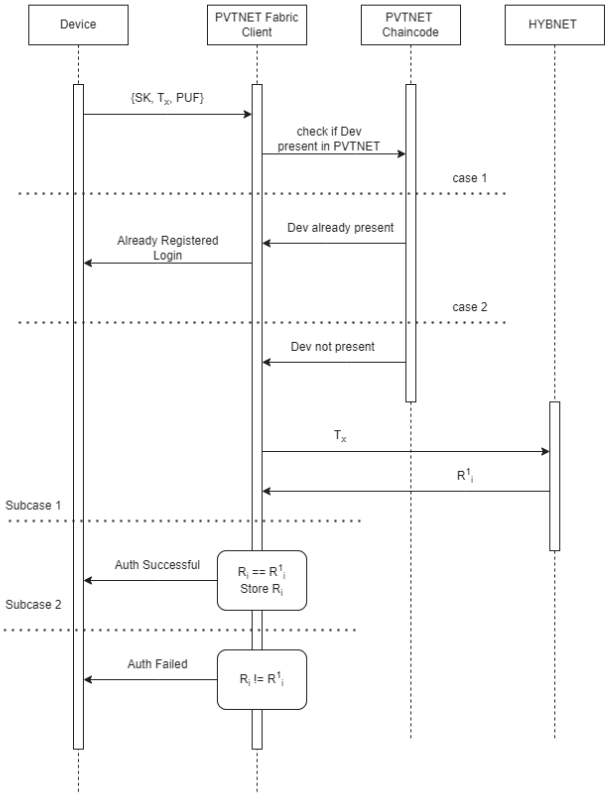


Fig. 3. Device Authentication

1) Setting up network and joining the peers: Setting up a blockchain network is discussed below. The steps for building blockchain will be same for both hybrid & private blockchain networks unless it is mentioned separately. The first step is to generate crypto material using cryptogen tool (available in Hyperledger fabric). This generates files required to up the containers of peers and orderers of all three organizations (Org1, Org2, Org3). The crypto material also includes the blockchain's genesis block, which is the first block of the chain, which is essential for the network to run. This material also contains the private key of the peers and orderer, certificates to authorize any user to join the network, and authority files to join any new authentic peers to the network to take part in consensus. The next step is to setup the containers of peers & orderer. Containers are started running using docker images of peer, orderer, and certificate authorities in Hyperledger fabric. Docker-compose configuration is written so that the orderer is exposed to communicate with the organizations in the network. The peers can interact with the orderer. Three CouchDB instances are started (one for each peer), they store the ledger sheet of the blockchain.

Once all the instances required for the network are setup, the instances are needed to be connected. Each CouchDB instance has already connected to its respective peer while instantiating the peer-containers. The peers' configuration is changed to make them communicate with the orderer running at a certain port. The orderer and the three peers of organizations Org1, Org2, and Org3 are added to the channel (communication medium). This channel can be considered a network. Once the organizations have joined the network, anchor-peers (peers responsible for gossiping, they communicate with the orderer and pass the information to other peers of the same organization) are updated. For the proposed protocol, we have used only one peer. As there is only a peer existing for each organization, each is an anchor peer. All the peers in the network are endorsing peers (peers participating in the consensus).

2) Chaincode: As the network is developed, the next step is to develop chaincode and deploy it onto the peers. Chaincode is coded in Go Language and contains Device Contract, where R_i of the device is stored in the ledger as a transaction. The ID of the transaction is the hash of R_i. CRUD (Create, Retrieve, Update, Delete) Operations are coded in the chaincode. The chaincode can fetch the transaction given the transaction hash as the input. The following steps are involved in deploying the chaincode onto the peers:

1) Installing all dependencies
2) Package Chaincode
3) Install Chaincode on all Endorsing Peer
4) Approve Chaincode as per Lifecycle Endorsement Policy
5) Commit Chaincode Definition

After the above-mentioned steps are done, the chaincode is initiated, and a sample input is given, which is considered as invoking (executing) the chaincode inorder to test it.

A binary called peer (Available in Hyperledger Fabric) handles joining, connecting peers and deploying chaincode onto the peers.

3) Fabric Client API services: A Fabric blockchain network can be interacted with by applications using the Hyperledger Fabric SDK (Software Development Kit). It offers a straightforward API so that programmers and users can easily add transactions to ledgers or query their contents. The Fabric Client API is an API that communicates with a private blockchain network. The API communicates with the certificate authorities in order to register a device or user to the network. By registering, API gets a private key, which is stored in the wallet of their respective organization. The certificates of each organization are generated using fabric tools and stored. The certificates, private key of devices are generated using their organisation's Membership Service Provider (MSP). Users can register to an organisation specifying the organisation name, Manufacturer ID. After getting registered, the user can login with their credentials. The user can invoke transactions here after a successful login.

V. Results and Analysis

Two Fabric Client applications are designed and developed. One for HYBNET and the other for PVTNET. The Fabric-client API of HYBNET provides end-points where it allows manufacturers to register and login. The logged-in manufacturer can add a device's R_i through a request which stores the R_i and returns a T_x for this device. The HYBNET stores the R_i of the device, and the ID of the asset in the chaincode is hashed value of R_i known as T_x. Fig.4 shows the processing of requests made to HYBNET and the registration of the device in HYBNET.

When a device tries to enter PVTNET, it has to communicate using Fabric Client API and get authenticated by following the proposed protocol. The Fig.5 shows successful registration of device in PVTNET which indicates successful authentication.

The Fabric client APIs are tested using Postman, is an API platform where developers can design, build, test and iterate their APIs. Fig.6 shows the working of API in Postman. The APIs allow you to register into it and login. In order to access the services provided in the application, the user or a device has to login. On successful login the device gets a JWT which is valid for only a certain amount of time. With the help of JWT, the device can access the services.

The PUF key generation and the proposed authentication protocol is implemented in the physical testbed as shown in Fig.7. The components used for the testbed setup is shown in the Table III.

TABLE III. Components used in experimental setup

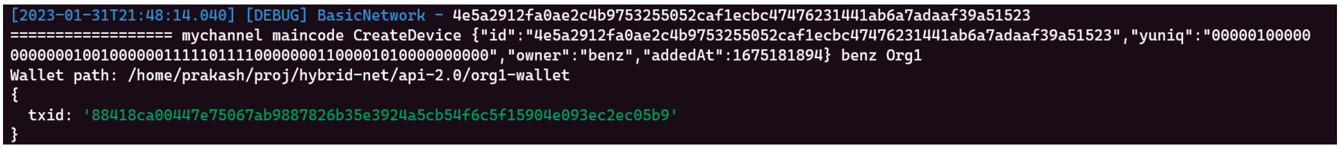
Component	Specification
Raspberry Pi	4 model B, 64-bit Quad core cortex-A72 (ARM V8) processor, 1.5 GHz clock speed, raspbian Operating system, 4 GB SDRAM
Arduino	Mega 2560 microcontroller, 16MHz clock speed, 8 KB SRAM, 256 KB flashmemory
Laptop	AMD Ryzen 7 processor, 2.9 GHz clock speed, Windows 11 Operating system with WSL 2, 16 GB RAM, 512 SSD

The time analysis of the proposed authentication protocol is shown in the Table IV.

A. Computational Cost Analysis

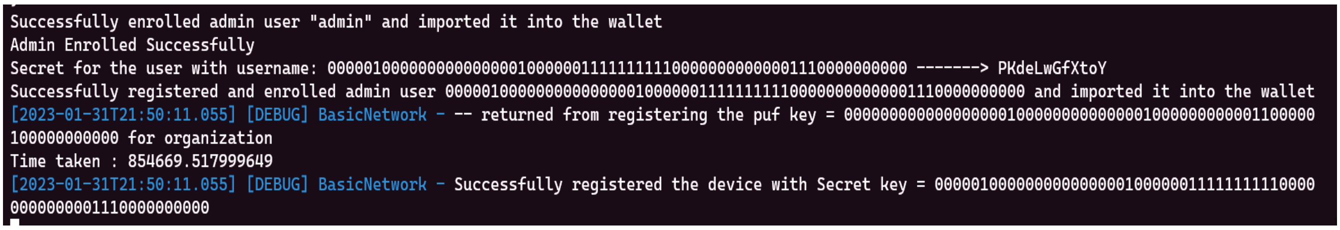
TABLE IV. Time Analysis of Protocol

Time parameters	Time Taken (μ sec)
Time to register a device in HYBNET	4555395.690
Time to authenticate a new device in PVTNET	160103.526
Time to authenticate a device that is already registered in PVTNET	3526.432



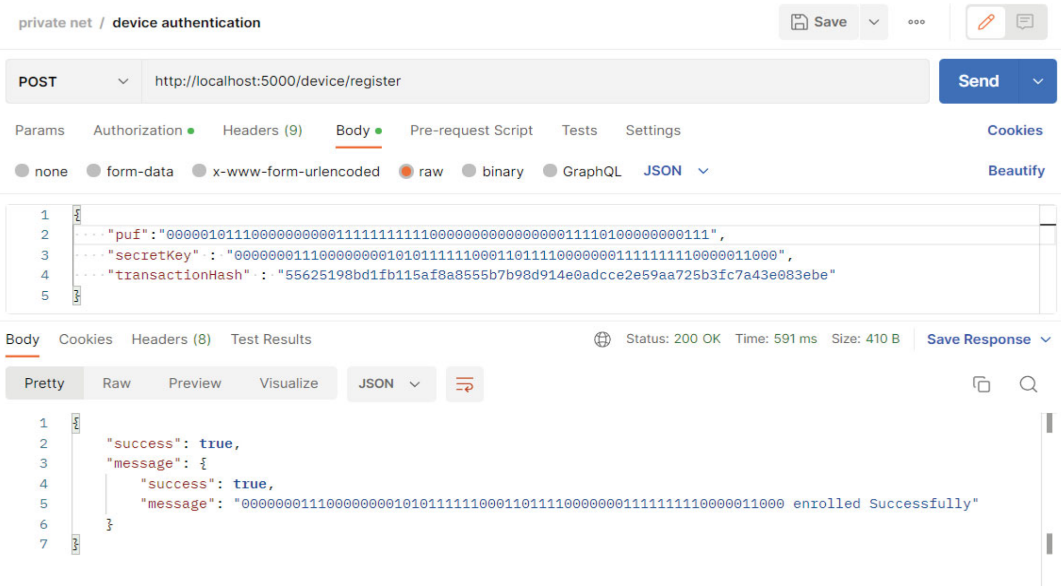
```
[2023-01-31T21:48:14.040] [DEBUG] BasicNetwork - 4e5a2912fa0ae2c4b9753255052caf1ecbc47476231441ab6a7adaaf39a51523
================= mychannel maincode CreateDevice {"id":"4e5a2912fa0ae2c4b9753255052caf1ecbc47476231441ab6a7adaaf39a51523","yuniq":"00000100000
000000010010000001111101111000000011000010100000000000","owner":"benz","addedAt":1675181894} benz Org1
Wallet path: /home/prakash/proj/hybrid-net/api-2.0/org1-wallet
{
  txid: '88418ca00447e75067ab9887826b35e3924a5cb54f6c5f15904e093ec2ec05b9'
}
```

Fig. 4. Registering device by Manufacturer



```
Successfully enrolled admin user "admin" and imported it into the wallet
Admin Enrolled Successfully
Secret for the user with username: 00000100000000000000001000000111111111100000000000000111000000000 -------> PKdeLwGfXtoY
Successfully registered and enrolled admin user 00000100000000000000001000000111111111100000000000000111000000000 and imported it into the wallet
[2023-01-31T21:50:11.055] [DEBUG] BasicNetwork - -- returned from registering the puf key = 0000000000000000001000000000000001000000000001100000
100000000000 for organization
Time taken : 854669.517999649
[2023-01-31T21:50:11.055] [DEBUG] BasicNetwork - Successfully registered the device with Secret key = 00000100000000000000001000000111111111110000
000000001110000000000
```

Fig. 5. Authentication of device in PVTNET



POST http://localhost:5000/device/register

○ none ○ form-data ○ x-www-form-urlencoded ● raw ○ binary ○ GraphQL JSON

```
{
    "puf":"00000101110000000000001111111111100000000000000000111101000000000111",
    "secretKey" : "00000001110000000001010111111000110111100000000111111110000011000",
    "transactionHash" : "55625198bd1fb115af8a8555b7b98d914e0adcce2e59aa725b3fc7a43e083ebe"
}
```

Body Status: 200 OK Time: 591 ms Size: 410 B

JSON

```
{
    "success": true,
    "message": {
        "success": true,
        "message": "00000001110000000001010111111000110111100000000111111110000011000 enrolled Successfully"
    }
}
```

Fig. 6. Authentication of the device using Postman

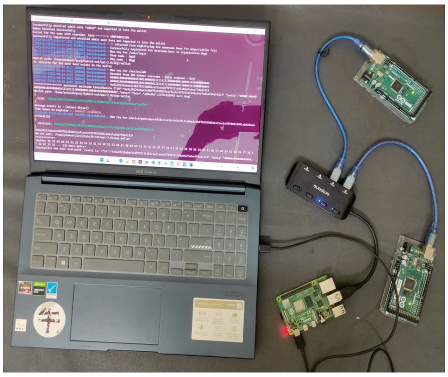


Fig. 7. Test bed to generate PUF key

The time taken to register a device in a Hybrid network is relatively high, as the registration in a Hybrid network includes the XOR operation to generate R_i, stored in the Hybrid blockchain network ledgers. The process also includes generation transaction hash T_x, which is stored in the device for further use. Compared the registration, authentication of a new device in PVTNET takes less time, as regeneration of PUF key and rendering of R_i from transaction hash T_x is involved, which needs to build a communication with the hybrid network and device. Finally the time taken to authenticate an already registered device in PVTNET is way less compared to registration and authentication as it involves minimum comparison steps.

The computational cost analysis of our proposed protocol is is shown in Table V. For computational cost analysis, we have considered Physical Uncontrollable Function (PUF), Hash (H), bitwise-XOR (XOR), Random number (R) and Bio-metric (B) as metrices. Each metric is denoted by a subscript, showing the number of times it is used in the protocol. The overall computational cost of our proposed protocol is very low when compared to other state-of-the-art protocols. The comparison is shown in the Fig.8.

TABLE V. COMPUTATIONAL COST ANALYSIS

Protocol	Computational Cost
Protocol-1 [18]	H_{22} + PUF_5 + XOR_{16} + R_3 + B_1
Protocol-2 [14]	H_7 + PUF_2 + XOR_{20} + R_7 + B_1
Protocol-3 [19]	H_{14} + PUF_2 + XOR_{16} + R_4 + B_1
Proposed Protocol	**H_2 + PUF_2 + XOR_2 + R_1**

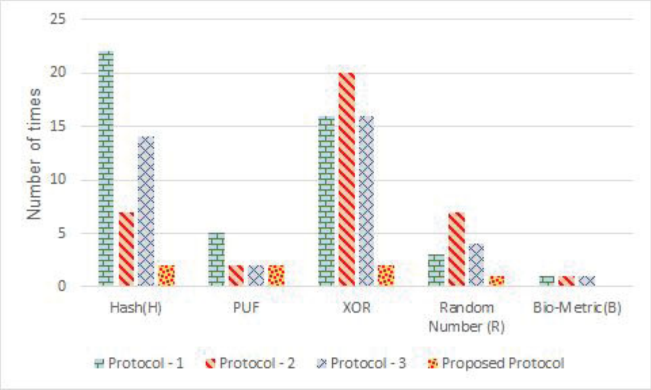


Fig. 8. Computational Cost Analysis

VI. CONCLUSION AND FUTURE WORK

Securing the IoT devices and data associated with the devices is a challenging task in the current era. The proposed authentication protocol secures IoT device and IoT device credentials and enables devices to access private network services securely using blockchain. The device-specific credentials like PUF key and secret key are used to verify the identity and authenticity of the device. The blockchain provides immutability to the device credentials and stores them securely. The proposed protocol is physically tested and implemented in a testbed. The proposed protocol is computationally less expensive when compared to other state-of-the-art protocols. The future work will be to add scalability to the proposed protocol and to do a formal security analysis of the same.

REFERENCES

[1] M. J. a. Mahmod and U. Guin, “A robust, low-cost and secure authentication scheme for iot applications,” *Cryptography*, vol. 4, no. 1, p. 8, 2020.

[2] U. Marjit and P. Kumar, “Towards a decentralized and distributed framework for open educational resources based on ipfs and blockchain,” in *2020 International Conference on Computer Science, Engineering and Applications (ICCSEA)*. IEEE, 2020, pp. 1–6.

[3] I. Fedorov, A. Pimenov, G. Panin, and S. Bezzateev, “Blockchain in 5g networks: Perfomance evaluation of private blockchain,” in *2021 Wave Electronics and its Application in Information and Telecommunication Systems (WECONF)*. IEEE, 2021, pp. 1–4.

[4] K. Y. Kamal and R. Muresan, “Mixed-signal physically unclonable function with cmos capacitive cells,” *IEEE Access*, vol. 7, pp. 130 977–130 998, 2019.

[5] G.-J. Schrijen and C. Garlati, “Physical unclonable functions to the rescue,” *Proceedings of the Embedded World*, 2018.

[6] C. Dannen, *Introducing Ethereum and solidity*. Springer, 2017, vol. 1.

[7] S. Nakamoto, “Bitcoin: A peer-to-peer electronic cash system,” *Decentralized Business Review*, p. 21260, 2008.

[8] B. K. Mohanta, S. S. Panda, and D. Jena, “An overview of smart contract and use cases in blockchain technology,” in *2018 9th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, 2018, pp. 1–4.

[9] D. Li, W. E. Wong, and J. Guo, “A survey on blockchain for enterprise using hyperledger fabric and composer,” in *2019 6th International Conference on Dependable Systems and Their Applications (DSA)*, 2020, pp. 71–80.

[10] W. F. Silvano and R. Marcelino, “Iota tangle: A cryptocurrency to communicate internet-of-things data,” *Future Generation Computer Systems*, vol. 112, pp. 307–319, 2020.

[11] Guartime and I. ID, “Internet of things authentication: A blockchain solution using sram physical unclonable functions,” 2017.

[12] O. N. Diedhiou and C. Diallo, “An iot mutual authentication scheme based on puf and blockchain,” in *2020 International Conference on Computational Science and Computational Intelligence (CSCI)*, 2020, pp. 1034–1040.

[13] V. K. Bathalapalli, S. P. Mohanty, E. Kougianos, B. K. Baniya, and B. Rout, “Pufchain 2.0: Hardware-assisted robust blockchain for sustainable simultaneous device and data security in smart healthcare,” *SN Computer Science*, vol. 3, no. 5, pp. 1–19, 2022.

[14] M. Masud, G. Gaba, P. Kumar, and A. Gurtov, “A user-centric privacy-preserving authentication protocol for iot-ami environments,” *Computer Communications*, vol. 196, pp. 45–54, 12 2022.

[15] K. P. Satamraju and B. Malarkodi, “A decentralized framework for device authentication and data security in the next generation internet of medical things,” *Computer Communications*, vol. 180, pp. 146–160, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0140366421003492

[16] D. Li, R. Chen, D. Liu, Y. Song, Y. Ren, Z. Guan, Y. Sun, and J. Liu, “Blockchain-based authentication for iiot devices with puf,” *Journal of Systems Architecture*, vol. 130, p. 102638, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1383762122001618

[17] L. Negka, G. Gketsios, N. A. Anagnostopoulos, G. Spathoulas, A. Kakarountas, and S. Katzenbeisser, “Employing blockchain and physical unclonable functions for counterfeit iot devices detection,” in *Proceedings of the International Conference on Omni-Layer Intelligent Systems*, 2019, pp. 172–178.

[18] P. Gope, A. K. Das, N. Kumar, and Y. Cheng, “Lightweight and physically secure anonymous mutual authentication protocol for real-time data access in industrial wireless sensor networks,” *IEEE Transactions on Industrial Informatics*, vol. 15, no. 9, pp. 4957–4968, 2019.

[19] W. Wang, Q. Chen, Z. Yin, G. Srivastava, T. R. Gadekallu, F. Alsolami, and C. Su, “Blockchain and puf-based lightweight authentication protocol for wireless medical sensor networks,” *IEEE Internet of Things Journal*, vol. 9, no. 11, pp. 8883–8891, 2022.